Docket No. 50325-0076 (Seq. No. 1417) PATENT

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

BEFORE THE BOARD OF PATENT APPEALS AND INTERFERENCES

In re Application of	:	Confirmation No.: 4033
Sunil K. Srivastava	:	Group Art Unit: 2131
Serial No.: 09/408,420	:	Examiner: Syed Zia
Filed: September 29, 1999	:	
For: METHOD FOR OVERCOMING THE SINGLE POINT OF FAILURE OF THE CENTRAL GROUP CONTROLLER IN A BINARY TREE GROUP KEY EXCHANGE APPROACH	:	

APPEAL BRIEF

Board of Patent Appeals and Interferences
United States Patent and Trademark Office
P.O. Box 1450
Alexandria, VA 22313-1450

Sir:

This Appeal Brief is submitted in support of the Notice of Appeal that was filed on May 6, 2005.

I. REAL PARTY IN INTEREST

Cisco Systems, Inc., which owns the assignee Cisco Technology, Inc., both of San Jose, California, are the real parties in interest.

II. RELATED APPEALS AND INTERFERENCES

Appellants are unaware of any related appeals and interferences.

III. STATUS OF CLAIMS

Claims 1-80 are pending in the application and were finally rejected in the Final Office Action mailed on February 10, 2004. Specifically, Claims 1-80 have been rejected under 35 U.S.C. § 103(a) as allegedly unpatentable over U.S. Patent Number 5,748,736 issued to Mittra in view of U.S. Patent Number 6,745,243 issued to Squire et al. It is from this final rejection of Claims 1-80 that this Appeal is taken.

IV. STATUS OF AMENDMENTS

The claims have not been amended after the Final Office Action.

V. SUMMARY OF CLAIMED SUBJECT MATTER

The present application contains independent Claims 1, 11, 21, 24, 31, 41, 51, 54, 61, 64, 71, and 74. The application and claims generally address problems with distributing keys for secure communication in a multicast network of routers, switches, etc.

Claims 1, 11, 31, and 41 recite similar limitations, except in the context of a method, a computer-readable medium, a computer system, and an apparatus, respectively. Claims 1, 11, 31, and 41 are directed to managing addition and deletion of network nodes to and from a secure multicast or broadcast group using multiple group controllers that are each a replica of a particular group controller and in which the network nodes and group controllers are logically organized in a binary tree. (Application, page 8, lines 3-8.)

For example, FIG. 6B depicts a group controller GC 501 that is responsible for a multicast group that includes users A-H, which means that GC 501 is responsible for managing the encryption keys used by the members of the group, such as distributing the keys to new group members when they join the group or updating the keys when a member leaves the group. (Application, page 35, line 12 – page 36, line 4.)

In the approach of Claim 1, multiple group controllers are used and each group controller is a replica of a particular group controller, such as in FIG. 10 that depicts group controllers GC, GC1, GC2, GC3, and GCN. (Application, page 37, line 19.) Each of group controllers GC1 through GCN depicted in FIG. 10 is a replica of group controller GC, which in turn is a group controller of the type depicted as GC 501 in FIG. 6B. (Application, page 37, line 22 – page 38, line 3; page 40, lines 14-15.) By using multiple group controllers that are

each a replica of a particular group controller, the single point of failure problem resulting from the use of a single central group controller is avoided. (Application, page 8, lines 20-21; page 37, lines 15-16.) In addition, the concern over using a single group controller that can result in a potential bottleneck is alleviated. (Application, page 6, lines 9-11.)

In addition, with the approach of Claim 1, the network nodes in the multicast or broadcast group and the replicated group controllers are logically organized according to a binary tree in which leaf nodes represent network nodes, intermediate nodes represent other network nodes, and root nodes represent the replicated group controllers, such as in the binary tree depicted in FIG. 5. Specifically, GC 501 in FIG. 5 is the root node that represents the replicated group controllers (e.g., GC, GC1, ..., GCN of FIG. 10A). (Application, page 31, lines 6-9.) Leaf nodes represent the members of the group, such as users A – H in FIG. 5. (Application, page 31, line 9.) Finally, intermediate nodes 515 represent other logical nodes that help to conceptually illustrate how the leaf nodes obtain group session keys (GKs) from the group controllers represented by the root node, GC 501. (Application, page 31, lines 10-13.)

In a binary tree as featured in Claim 1, examples of which are depicted in FIG. 5, FIG. 6B, and FIG. 7B, the nodes of the binary tree are organized such that each node has one parent node (except for the root node) and two child nodes. Thus, at each node of the binary tree, there are at most two branches, hence the name "binary" tree. This is in contrast to a general, non-binary hierarchical tree, such as that depicted in FIG. 9, in which node GC 901 has eight child nodes. Therefore, because there are more than two child nodes of node GC 901, FIG. 9 does not depict a binary tree.

In the approach of Claim 1, a first group controller is joined to the plurality of group controllers in a local network, such as by joining group controller GC of FIG. 10A to network 1004 that includes group controllers GC1-GCN, each of which is a replica of group controller GC, as depicted by block 1010 of FIG. 10B. (Application, page 40, lines 12-15.) Then using a key exchange protocol, a secure communications channel is established between the first group controller and a second group controller of the plurality of group controllers, such as by building a secure channel using Diffie-Hellman key exchange between controller GC and another other replica group controller of GC1-GCN of FIG. 10A, as depicted by block 1020 of FIG. 10B. (Application, page 40, lines 16-19.)

In the approach of Claim 1, the secure communications channel is established between two of the replicated group controllers (e.g., between GC and GC3 of FIG. 10A) instead of between a group controller and a member of the group that is controlled by the group controllers (e.g., not between group controller GC of FIG. 10A and group member user C of FIG. 6B).

Next, a request from a load balancer to add or delete a network node of the secure multicast or broadcast group is received, such as receiving a request from load balancer 1002 of FIG. 10A for a node to join or leave group, and in response the load balancer distributes the request to one of group controllers GC, GC1, ..., GCN, as depicted by block 1013 of FIG. 10B. (Application, page 40, line 20 - page 41, line 2.)

When the group controller that receives the request is designated as the master group controller (block 1014 of FIG. 10B), that group controller generates a new group session key for use by the nodes in the binary tree that are affected by the joining or leaving node, such as the nodes in leaf branch 621 of FIG. 6B, as depicted by block 1016 of FIG. 10B. (Application, page 41, lines 7-12.) That new group session key is distributed to the other replicated group controllers, and from them the group key is distributed to the network nodes, as indicated by block 1018 of FIG. 10B. (Application, page 41, lines 12-14.)

The approaches of Claims 11, 31, and 41 are supported by the same portions of the application as described above with respect to Claim 1. Specifically, Claim 11 is a computer-readable medium claim that features the same steps as in Claim 1. In addition, Claim 11 being directed to a computer-readable medium that carries sequences of instructions that cause on one or more processors to carry out the recited steps is supported by the "HARDWARE OVERVIEW" section of the application.

In particular, main memory 807 of FIG. 8 stores instructions for execution by processor 805. (Application, page 42, lines 1-3; page 43, lines 1-2.) Such instructions can be read into main memory 807 from a computer-readable medium, such as storage device 811, in which execution of the sequences of instructions contained in main memory 807 causes the processor 805 to perform the process steps described in the application. (Application, page 42, lines 3-6.) The term "computer-readable medium" refers to any medium that participates in providing instructions to processor 805 for execution, including but not limited

to, non-volatile media, volatile media, and transmission media. (Application, page 43, lines 11-14.)

Regarding Claim 31, which is in the context of a computer system, the above discussion with respect to method Claim 1 and computer-readable medium Claim 11 applies as Claim 31 includes a memory with sequences of instructions and one or more processors that execute the instruction to carry out the recited steps. In addition, Claim 31 includes a load balancer and a bus, such as load balancer 1002 of FIG. 10A or load balancer 205 of FIG. 2A and bus 803 of FIG. 8.

Regarding Claim 41, which is in the context of an apparatus, the above discussion with respect to Claims 1, 11, and 31 applies as Claim 41 includes means plus function elements (identified by the elements that being "means for...") for performing the functions of the steps recited in Claims 1, 11, and 31. In particular, a group controller such as GC 501 of FIG. 5 or GC, GC1, ..., GCN of FIG. 10A, can be used to implement the functions recited in the steps of Claims 1, 11, and 31. In addition, FIG. 2A depicts a group controller 201 that can be implemented as a clustered central key distribution center (KDC) in the form of a "server farm" comprising multiple KDC servers 201a-201d. (Application, page 18, lines 10-12.)

Claims 21, 51, 61, and 71 recite similar limitations, except in the context of a method, a computer-readable medium, a computer system, and an apparatus, respectively. Claims 21, 51, 61, and 71 are directed to managing addition and deletion of network nodes from and to a secure multicast or broadcast group using multiple group controllers that include replicated information and in which the network nodes and group controllers are logically organized based on a binary tree. In particular, the features of Claims 21, 51, 61, and 71 parallel and are similar to those of Claims 1, 11, 31, and 41, and therefore such features will not be discussed again herein.

However, despite the similarities between Claims 21, 51, 61, and 71 and Claims 1, 11, 31, and 41, there are differences in the approach of Claims 21, 51, 61, and 71 as compared to Claims 1, 11, 31, and 41. For example, Claims 21, 51, 61, and 71 feature a first group controller that comprises information that is replicated in the plurality of group controllers, and thus the first group controller of Claims 21, 51, 61, and 71 is similar to the "particular group controller" of Claims 1, 11, 31, and 41. Also, the secure channel is

established between the first group controller and the plurality of group controllers in Claims 21, 51, 61, and 71, instead of between a first group controller and a second group controller as in Claims 1, 11, 31, and 41. Also, the new group session key is merely "generated" in Claims 21, 51, 61, and 71, as compared to being "created and stored" in Claims 1, 11, 31, and 41. Finally, in the last step, the group session key is distributed from the first group controller to the other group controllers over the secure channel in Claims 21, 51, 61, and 71, instead of the group session key being distributed from a third group controller to the network nodes as in Claims 1, 11, 31, and 41.

Claims 24, 54, 64, and 74 recite similar limitations, except in the context of a method, a computer-readable medium, a computer system, and an apparatus, respectively. Claims 24, 54, 64, and 74 are directed to creating a secure multicast or broadcast group using multiple group controllers and a logical arrangement of the network nodes in a binary tree structure in which the group controllers correspond to the root node of the binary tree. In particular, the features of Claims 24, 54, 64, and 74 parallel several of those of Claims 1, 11, 31, and 41, and therefore such features will not be discussed again herein.

However, despite the similarities between Claims 24, 54, 64, and 74 to Claims 1, 11, 31, and 41, there are differences in the approach of Claims 24, 54, 64, and 74 as compared to Claims 1, 11, 31, and 41. For example, Claims 24, 54, 64, and 74 omit the joining, receiving, and creating and storing steps of Claims 1, 11, 31, and 41, plus Claims 24, 54, 64, and 74 do not include a feature similar to the group controllers being a replica of a particular group controller. In addition, Claims 24, 54, 64, and 74 include an additional feature not found in Claims 1, 11, 31, and 41, namely load balancing the traffic from the network nodes among the plurality of group controllers, such as by using a load balancer as featured in Claims 1, 11, 31, and 41 and discussed above. Finally, the group session key is distributed by one of the group controllers, but unlike in Claims 1, 11, 31, and 41 where the group session key is distributed to the plurality of group controllers, the entities to which the group session key is distributed is not specified in Claims 24, 54, 64, and 74.

VI. GROUNDS OF REJECTION TO BE REVIEWED ON APPEAL

Claims 1-80 have been rejected under 35 U.S.C. § 103(a) as allegedly unpatentable over U.S. Patent Number 5,748,736 issued to Mittra (" *Mittra* ") in view of U.S. Patent Number 6,745,243 issued to Squire et al (" *Squire* ").

VII. ARGUMENT

A. Introduction

"To establish a prima facie case of obviousness [under 35 U.S.C. §103(a)] of a claimed invention, all the claim limitations must be taught or suggested by the prior art." MPEP §2143.03. The Appellants respectfully submit that neither *Mittra* nor *Squire*, either alone or in combination, disclose, teach, suggest, or render obvious all the limitations of Claims 1-80. In the Final Office Action's citations to *Mittra* and *Squire*, the Examiner has repeatedly mischaracterized the disclosures, and the features of the claims that the Examiner alleges are disclosed are simply not present in either *Mittra* or *Squire*. Therefore, the Examiner has failed to establish a prima facie case of obviousness because many of the claim limitations are neither taught nor suggested by the prior art.

In particular, for at least the reasons set forth herein, neither *Mittra* nor *Squire* disclose anything relating to "**a plurality of group controllers**" as featured in Claims 1-80. (Emphasis added.) *Mittra* only discloses a *single* group security controller, *Squire* discloses nothing about a group controller, and therefore neither *Mittra* nor *Squire* disclose "a plurality of group controllers."

Furthermore, for at least the reasons set forth herein, neither *Mittra* nor *Squire* disclose anything relating to "**each group controller** of the plurality of group controllers **is a *replica*** of a particular group controller" as featured in Claims 1-20 and 31-50, "a first group controller comprising information that is replicated in a plurality of group controllers" as featured in Claims 21-23, 51-53, 61-63, and 71-73, or "circulating a token among the plurality of group controllers to designate the one group controller as having permission to selectively generate the group session key" as featured in Claims 25, 55, 65, and 75. (Emphasis added.) *Mittra* nor *Squire* simply do not disclose anything about multiple group controllers that are each a ***replica*** of a particular group controller.

In addition, for at least the reasons set forth herein, neither *Mittra* nor *Squire* disclose anything relating to a logical arrangement of network nodes and group controllers based on "a ***binary*** tree" or that the "**root** node" of the binary represent or corresponds to the "plurality of group controllers" as featured in Claims 1-80. (Emphasis added.) While *Mittra* discloses a hierarchical diagram of the system described herein, the disclosed hierarchy is simply not a binary tree, and *Squire* simply discloses nothing that can be construed as any type of tree, little less a binary tree.

Finally, for at least the reasons set forth herein, neither *Mittra* nor *Squire* disclose anything relating to "**receiving a request** to **add or delete a network node** of the secure multicast or broadcast group **from a *load balancer***" as featured in Claims 1-23, 31-53, 61-63, and 71-73 or "**load balancing traffic** emanating **from** a plurality of **network nodes to** the plurality of **group controllers**" as featured in Claims 24-30, 54-60, 64-70, and 74-80. (Emphasis added.) While *Squire* discloses a device that includes both network address translation and load balancing, nothing in *Squire* discloses that such a device receives requests to change the member of a multicast, and *Mittra* discloses nothing at all about load balancing.

B. Claim 1 Is Patentable Over *Mittra* in View of *Squire*

(1) INTRODUCTION TO CLAIM 1

Claim 1 features:

> "A method for managing addition and deletion of network nodes from and to a secure multicast or broadcast group of network nodes in a communications network without a single point of failure, wherein each of the network nodes is associated with one of a ***plurality*** **of group controllers**, wherein **each group controller** of the plurality of group controllers **is a *replica*** of a particular group controller, and wherein the network nodes and the plurality of group controllers are **logically organized in a *binary* tree** that represents the network nodes and the plurality of group controllers, in which leaf nodes of the binary tree represent network nodes that are joining or leaving the secure multicast or broadcast group, intermediate nodes represent other network

nodes, and ***root*** **nodes represent the plurality of group controllers**, the method comprising the steps of:

joining a **first group controller** to the plurality of group controllers in a local network;

establishing a secure communication channel between the **first group controller** and a **second group controller** of the plurality of group controllers using a key exchange protocol;

receiving a request to **add or delete a network node** of the secure multicast or broadcast group **from a *load balancer*** that is coupled to the plurality of group controllers;

creating and storing a new group session key for each network node represented in each branch of the binary tree that is affected by adding or deleting the network node from the secure multicast or broadcast group; and

distributing a group session key from a third group controller of the plurality of group controllers to the network nodes." (Emphasis added.)

Thus, Claim 1 features an approach for managing the addition and deletion of network nodes for a secure multicast or broadcast group *without a single point of failure* that includes the following features: (a) a ***plurality of group controllers***; (b) each group controller of the ***plurality of group controllers*** is a **replica** of a particular group controller; (c) a **binary tree**; (d) ***root nodes*** of the binary tree that represent the plurality of group controllers; and (e) **receiving a request** to **add or delete a network node** of the secure multicast or broadcast group **from a *load balancer***.

Note that in Claim 1, a plurality of group controllers is featured, which by definition means **two or more group controllers**, and each group controller in the plurality of group controllers is a **replica** of a particular group controller. In particular, Claim 1 recites three group controllers of the plurality of group controllers, namely the first group controller that is joined to the plurality of group controllers, a second group controller for which a secure communications channel is established to the first group controller, and a third group controller from which the group session key is distributed to the plurality of group controllers. The use of a plurality of group controllers, *instead of a single* group controller, addresses the drawback of a single group controller being a potential bottleneck (Application,

page 6, lines 9-11) and the issue that a central group controller presents a single point of failure (Application, page 7, lines 7-9).

Also note that, by definition, in a binary tree, each node contains one parent node and no more than two child nodes, as exemplified in the embodiments illustrated in FIG. 5, FIG. 6B, and FIG. 7B of the Application.

(2) INTRODUCTORY DISCUSSION OF *MITTRA* AND *SQUIRE*

In contrast to the approach of Claim 1, *Mittra* discloses an approach for secure group communication via a multicast or broadcast transmission that uses ***one and only one*** group security controller (GSC) and at least one trusted intermediary (TI) server. In particular, *Mittra* notes that the "inventive 'secure multicast' implement by the FIG. 1 system is controlled by a ***single*** group security controller (GSC 111), and the inventive 'secure multicast' group implemented by each of the systems of FIGS. 2 and 3 is controlled by a ***single*** group security controller (GSC 11 or 211)." (Col. 6, lines 62-67; emphasis added.) Note that each of Figures 1-3 of *Mittra* shows ***one and only one*** GSC.

The approach of *Mittra* is vulnerable to the single point of failure problem due to the use of just a ***single*** group security controller. Also note that while *Mittra* is described in terms of a secure multicast group having a hierarchical structure (Col. 7, line 1; Figures 1-3), the ***hierarchical structure is not in the form of a binary tree*** because many nodes have more than two child nodes (e.g., in Figure 1, multicast/unicast network 112A has six child nodes and multicast/unicast networks 112B and 112D each have three child nodes; Figures 2 and 3 include corresponding examples of three or more child nodes).

Also, *Mittra* describes the use of trusted intermediary (TI) servers, explaining that each TI server "is a special type of sender and receiver. The TI servers create a (logical) hierarchy of secure multicast networks (the secure distribution tree) that makes the system of the invention scalable (able to practically implement a group of any number of members)." (Col. 4, lines 20-25.) The TI servers "act as proxies for the GSC" by changing group membership for the sub-group served by a particular TI server, and thus the burden of managing group membership is spread between the GSC and the TI servers. (Col. 12, line 60 – Col. 13, line 3.)

Thus, in *Mittra*, each TI server is a merely a member of the group, albeit a "special" member that is entrusted with the ability to perform some group management functions, but only for the members below itself in the group hierarchy. This reduces the workload on the single GSC, thereby allowing for better scaling of the group to larger numbers of members. However, a TI server is not a group controller because the TI server cannot affect group members above itself in the group hierarchy. Also, if the TI servers were group controllers, *Mittra* would not describe the TI servers as "a special type of sender and receiver" but would instead describe them as group security controllers. The choice of name – trusted intermediary – also indicates that the TI servers are not group security controllers. As a result, the only group security controller in the approach of *Mittra* is the single GSC, and without the single GSC, the group using the approach of *Mittra* fails because only the GSC can create a group key for the entire group and affect membership for the entire group.

Also in contrast to the approach of Claim 1, *Squire* merely discloses an approach for network caching and load balancing employing network translation. (Col. 1, lines 19-20.) Specifically, the approach of *Squire* is directed to the problem of being able to discriminate network traffic based on network session information, such as at OSI layer 5, as a means for selecting traffic to cached or to determine network routing. (Col. 2, lines 53-60.) *Squire* describes FIG. 5 as illustrating a network device 500 that is endowed with the integrated network address translation (NAT)/network cache and/or NAT/load balancer described therein. (Col. 7, lines 9-29.) *Squire* explains that network device 500 includes a controller 502 that discriminates network traffic for caching and/or load balancing based on network session information identified by network address translator 501. (Col. 7, line 52 to Col. 8, line 3.) But nowhere in *Squire* is controller 502 described as having any role in the context of a multicast or broadcast group, little less the role of a group controller in a secure multicast or broadcast.

In summary, the Examiner has failed to establish a prima facie case of obviousness because *Mittra* and *Squire* do not teach numerous limitations of the claims. The Examiner has repeatedly mischaracterized the disclosures of the prior art and summarily stated that a disclosure of a feature "X" is the same as a claimed feature "Y" when the features "X" and "Y" are clearly not the same. Furthermore, in the Final Office Action, the Examiner has not given any rationale why the Examiner believes the disclosed features are the same as the

claimed features or even why the latter would be obvious in light of the former. The few insights into the Examiner's reasoning that the Appellant has been able to discern from the Final Office Action and the Examiner Interview are either unsupported by the prior art or simply illogical.

(3) NEITHER *MITTRA* NOR *SQUIRE* DISCLOSE A "PLURALITY OF GROUP CONTROLLERS"

(A) INTRODUCTION

The approach of Claim 1 involves a "***plurality of group controllers***," yet *Mittra* only discloses only ***one*** group security controller (GSC), and *Mittra* emphasizes with respect to Figures 1-3, that only a "***single***" group security controller is used. (Col. 6, lines 62-67; emphasis added.) Each of Figures 1-3 in *Mittra* discloses only ***one*** GSC (e.g., GSC 11 in Figure 1, GSC 11 in Figure 2, and Sender/GSC 211 in Figure 3).

Mittra explains that "all that is required to begin secure multicast is that **the** GSC is started up. Once this is done, senders and receivers apply to join the group as described below." (Col. 7, lines 32-34; emphasis added). "Joining a secure multicast group requires the joining member first to set up a separate secure channel with **the** GSC of the group (using a unicast communication line)." (Col. 7, lines 45-47; emphasis added). "Only **the** GSC maintains information concerning group membership; members do not know about each other..." (Col. 7, lines 64-65; emphasis added). *Mittra* then describes the set up of the secure channel (Col. 8, lines 3-14), then the communications between the GSC and the new member (Col. 8, lines 15-22), the communication of the new Kgrp to the multicast and the new member (Col. 8, lines 23-35), and then two cases for handling a member leaving the group (Col. 8, lines 36-67). Thus, *Mittra* fails to disclose anything other than a ***single*** group security controller.

Mittra distinguishes between the single GSC and the members of the multicast when *Mittra* explains that there "are ***three types of entities*** participating in a secure multicast: senders, receivers, and a group security controller (GSC). The single GSC (e.g., GSC 111 of FIG. 1) provides key management and thus effectively controls [the] secure multicast group ("the group") membership. As far as the system is concerned, all that is required to begin a

secure multicast is that the GSC is started up. Once this is done, senders and receivers apply to join the group..." (Col. 7, lines 28-35.)

Thus, *Mittra* expressly distinguishes between the single GSC that controls the multicast group and the members of the multicast group, such as the senders and receivers as well as the TI servers that are special senders and receivers. If the TI servers were in fact group controllers, *Mittra* would refer to them as such, and the fact that *Mittra* describes the TI servers as special members indicates that the TI servers are not group controllers. Furthermore, Figures 1-3 of *Mittra* illustrate the single GSC at the top of the hierarchical structures depicted therein, not a group of GSCs, with the senders and receivers denoted as nodes labeled "S" and "R," respectively, and the TI servers labeled as "TI." Thus, the top most or **root node** in each of Figures 1-3 in *Mittra* represent only the ***single*** GSC.

(B) THE FINAL OFFICE ACTION'S CITATIONS RELATING "A PLURALITY OF GROUP CONTROLLERS" OF CLAIM 1

The Final Office Action cites "Fig. 1-3, col. 6 line 4 to line 45" as allegedly disclosing the features of the preamble of Claim 1, which includes the introduction of "a plurality of group controllers" and then includes four additional references to "the plurality of group controllers." However, as explained above, Figures 1-3 of *Mittra* disclose only ***one*** GSC (e.g., GSC 11 in Figure 1, GSC 11 in Figure 2, and Sender/GSC 211 in Figure 3). The cited portion of the description of *Mittra*, which cites nearly all of Column 6 without any explanation as to which portions of Column 6 allegedly disclosure the features of Claim 1, are the descriptions of Figures 1-3 that refer to "***a*** group security controller (GSC 111)" (Col. 6, lines 4-5; emphasis added) and "a combined group security controller and sender (GSC 211)" (Col. 6, lines 39-40; emphasis added.)

Thus, none of the portions of *Mittra* that the Examiner relies upon as allegedly disclosing "a plurality of group controllers" as featured in Claim 1 disclose anything other than a single group security controller, and therefore the Examiner has failed to establish a prima facie case of obviousness because this feature of Claim 1 is neither taught nor suggested by *Mittra*.

(C) THE FINAL OFFICE ACTION'S CITATIONS RELATING TO THE "JOINING" STEP OF CLAIM 1

Regarding the "joining" step of Claim 1 that features the first of three group controllers of the plurality of group controllers, the Final Office Action's citation from *Mittra* merely describes the "joining" of ***member*** to the secure multicast group, not the joining of a group controller. Specifically, *Mittra* explains: "Joining a secure multicast group requires the **joining** ***member*** to first set up a separate secure channel with the GSC of the group" and that the "purpose of the secure channel is to facilitate and isolate confidential communication between the **GSC** and this ***member*** during the time that the member is part of the group." (Col. 7, lines 45-51; emphasis added.)

However, Claim 1 features "joining a first group controller to the plurality of group controllers in a local network," which requires at least three group controllers, the first group controller that is joining the plurality of group controllers, which includes two or more group controllers. The cited portion of *Mittra* relied upon in the Final Office Action only discloses the single GSC and describes the joining of a ***member***, which could be either a sender or a receiver or even a special member such as a TI server to the group. But a ***member*** of a multicast group is different than the ***group security controller*** that manages the multicast group. The cited portion of *Mittra* discloses nothing about the joining of a ***group controller*** to the plurality of group controllers nor anything about more than one group controller, as featured in Claim 1, and therefore the Examiner has failed to establish a prima facie case of obviousness because this feature of Claim 1 is neither taught nor suggested by *Mittra*.

In the portion of the Office Action addressing the "joining" step, the Examiner has confused the members of the multicast group as disclosed in *Mittra* with the group controllers of Claim 1 that manage, or control, the multicast group. This mischaracterization of a multicast member as a group controller is apparent in several other portions of the Final Office Action, such as discussed below with respect to the "establishing" and "distributing" steps of Claim 1. However, in the approach of Claim 1, the plurality of group controllers that manages the multicast or broadcast group are fundamentally different than the "network nodes" that are the members of the multicast or broadcast group.

Furthermore, the difference between the entity managing a multicast group and the members of the multicast group is clearly apparent in *Mittra* that distinguishes between the

single GSC and the two types of members of the multicast. Specifically, *Mittra* explains that there "are ***three types of entities*** participating in a secure multicast: senders, receivers, and a group security controller (GSC). The single GSC (e.g., GSC 111 of FIG. 1) provides key management and thus effectively controls [the] secure multicast group ("the group") membership. As far as the system is concerned, all that is required to begin a secure multicast is that the GSC is started up. Once this is done, senders and receivers apply to join the group..." (Col. 7, lines 28-35.) Thus, the senders and receivers are the members of the multicast group that join or leave the group, whereas the group security controller is the entity that manages the membership of the group. Therefore, as in Claim 1, *Mittra* makes clear that the members of the group are fundamentally different than the group security controller that manages the group.

(D) THE FINAL OFFICE ACTION'S CITATIONS RELATING TO THE "ESTABLISHING" STEP OF CLAIM 1

Regarding the "establishing" step of Claim 1 that features both the first and second group controllers of the plurality of group controllers, the Final Office Action's citation from *Mittra* merely describes that after the GSC receives and approves a join request, the joining member's identification and information concerning the secure channel is added to a private database maintained by the GSC. (Col. 7, lines 52-54.) *Mittra* explains that the GSC therefore has full knowledge of the group membership and can communicate with each member over the secure channels, and that the member also stores the information for the secure channel for later communicating with the GSC. (Col. 7, lines 55-59.) Thus, the secure channel in *Mittra* is between the GSC and a ***member*** of the group.

However, Claim 1 features "establishing a secure communications channel between the first **group controller** and a second **group controller** of the plurality of group controllers using a key exchange protocol." (Emphasis added.) Yet nothing in the cited portion of *Mittra* discloses anything about a secure communications channel between two group controllers, only a secure channel between the GSC and a member of the group, such as a sender, a receiver, or a TI server. Therefore, the Examiner has failed to establish a prima facie case of obviousness because this feature of Claim 1 is neither taught nor suggested by *Mittra*.

(E) THE FINAL OFFICE ACTION'S CITATIONS RELATING TO THE "DISTRIBUTING" STEP OF CLAIM 1

Regarding the "distributing" step of Claim 1 that features a third group controller of the plurality of group controllers, the Final Office Action's citation from *Mittra* merely describes a key distribution scheme for the GSC to provide a new group key (e.g., Kgrp) to the ***members*** of the group. (Col. 8, lines 51-52.) *Mittra* then describes several ways this can be accomplished, such as by using the separate secure channels between the GSC and the group members or by using a single update message that includes different copies of the new group key, each copy being encrypted with the key for the secure channel for each member. (Col. 8, lines 53-65.) Thus, it is the same single GSC disclosed in the earlier portions of *Mittra* cited in the Final Office Action regarding the "joining" and "establishing" steps that is also distributing the new group key to the group members.

However, Claim 1 features "distributing a group session key from a third group controller of the plurality of group controllers to the network nodes." The third group controller is in addition to the first and second group controllers from the "joining" and "establishing" steps, yet *Mittra* discloses only a single GSC that is referred to in each of the Final Office Action's citations to *Mittra*. Therefore, the Examiner has failed to establish a prima facie case of obviousness because this feature of Claim 1 is neither taught nor suggested by *Mittra*.

(F) ADDITIONAL FINAL OFFICE ACTION CITATIONS FROM CLAIMS 21, 31, 41, 51, 61, AND 71 RELATING TO STEPS THAT ARE SIMILAR TO THOSE OF CLAIM 1

In the rejections of the "joining," "establishing," and "distributing" steps of Claims 21, 31, 41, 51, 61, and 71, the Final Office Action includes both the citations to *Mittra* discussed above, and additional citations that are discussed below and are therefore included here as part of the discussion of Claim 1.

Regarding the "joining" steps of Claims 21, 31, 41, 51, 61, and 71, the Final Office Action's additional citation from *Mittra* merely describes how a trusted intermediary (TI) can be used by members to join the group instead of contacting the GSC. (Col. 13, lines 39-44.) *Mittra* explains that TI's are used to create a logical hierarchy within the inventive secure multicast group, such as that depicted by the "TI" nodes of Figures 1-3 of *Mittra*. (Col. 12,

lines 29-38.) The TI servers act as proxies for the GSC by approving changes in group membership in the sub-group served by the TI. (Col. 12, lines 60-66.) Thus, a TI server acts as the GSC for the member of "its" subgroup. (Col. 12, line 67 – Col. 13, line 3.) Each TI server multicasts to the members of its group, whereas changes in membership of a top level group is handled by the GSC. (Col. 13, lines 8-10.) When the multicast begins, following the startup of the GSC, the TI servers each apply to join the group, just as the receiver and sender group members apply to join the group. (Col. 13, lines 26-27.) Finally, *Mittra* explains: "Each TI server is a trusted intermediary, which is a ***special* type *of sender and receiver***." (Col. 4, lines 20-21; emphasis added.)

Thus, a TI server is merely a group member, albeit a special group member, that can manage membership of other group members below the TI in the hierarchy, yet there remains a ***single*** GSC at the top or root node of the hierarchy. Therefore, a TI server is not a group controller because the TI server can only approve membership changes of a limited number of group members and because the TI server can only establish a subgroup key for the members of the TI server's subgroup, not a group key for the entire group. Only the single GSC is a group controller because only the single GSC can approve membership changes for the entire group and because only the single GSC can establish a new group key for the entire group. As a result, nothing in this additional cited portion of *Mittra* discloses a plurality of group controllers, as featured in Claim 1.

Regarding the "establishing" steps of Claims 21, 31, 41, 51, 61, and 71, the Final Office Action's additional citation from *Mittra* merely describes how a joining member can contact a parent TI server instead of the GSC with the TI server authenticating the joining member on behalf of the GSC. (Col 13, lines 45-53.) Thus, the only disclosed communication is between the TI server and the joining member, which means any channel established between the TI server and the joining member is between two members of the group, not between two group controllers as featured in Claim 1. Therefore the additional cited portion of *Mittra* discloses nothing about establishing a secure communications channel between two group controllers, little less disclosing anything about a plurality of group controllers, as featured in Claim 1.

Regarding the "distributing" steps of Claims 21, 31, 41, 51, 61, and 71, the Final Office Action's additional citation from *Mittra* merely describes how a member leaves a

group using TI's and how multicast transmissions are sent with TI's. (Col. 13, line 60 – Col. 14, line 10.) In particular, *Mittra* explains that while a TI server handles a leaving member in place of the GSC, a "TI server also has the option of leaving the group itself if it has no children interested in the multicast." (Col. 13, lines 65-67.) The ability of the TI server to leave the group is another difference between the TI servers and the GSC that illustrates that a TI server is not a group controller. Specifically, if the GSC were disassociate itself from the group members, the group would not be able to function, whereas a TI can leave the group if there are no members below the TI in the hierarchy.

In addition, the remainder of the additional cited portion of *Mittra* describing how multicast transmissions are sent using TI servers explains that if a sender is not at the top level multicast group, the sender's message is unicast up to a TI server, and then to another TI server, and so on until the top level group is reached, from where the message can then be multicast to all members of the group. (Col. 14, lines 1-10.) This portion of *Mittra* again illustrates that a TI server is not a group controller, since a TI server can only communicate with its parent and the group members below it, whereas the GSC is able to communicate with all members of the group.

Therefore, the additional cited portion of *Mittra* regarding the "distributing" step neither discloses or suggests anything about a third group controller distributing a group session key, little less a plurality of group controllers as featured in Claim 1, since TI servers are not group controllers because TI servers only communicate with those group members that are below the TI server in the hierarchy and because the TI servers have the ability to leave the group when there are no more children below the TI server. Therefore, the Examiner has failed to establish a prima facie case of obviousness because these features of Claim 1 are neither taught nor suggested by *Mittra*.

(H) EXAMINER INTERVIEW

The Appellant conducted an interview with the Examiner on April 21, 2005 that covered the arguments discussed above. In response to the argument that *Mittra* only discloses a single GSC instead of "a plurality of group controllers" as featured in Claim 1, the Examiner stated that a member of the multicast in *Mittra*, such as a sender or a receiver, could also act as a group controller. However, the Examiner did not cite any portion of

Mittra that supported this argument, and the Appellant has been unable to locate any such teaching in *Mittra*. Furthermore, the Appellant explained to the Examiner that the argument that a sender or receiver could act as a group controller contradicted the clear descriptions of Figures 1-3 of *Mittra* that repeatedly depict and describe only a "***single***" group security controller, and therefore the Examiner's assertion that a group member can act as a group controller is neither supported nor logical in light of the disclosure of *Mittra*.

Towards the end of the interview, the Examiner stated that the plurality of group controllers could be more clearly distinguished over *Mittra* if Claim 1 was amended to recite a description of the plurality of group controllers, such as by reciting a feature or product name for the "invention" or perhaps even by amending Claim 1 to recited a trademark used to refer to the "invention" that would not be found in a subsequent search by the Examiner. The Appellant replied that the recitation of a plurality of group controllers was sufficient to distinguish Claim 1 over *Mittra*, because *Mittra* only discloses a ***single*** GSC.

(G) CONCLUSION OF DISCUSSION THAT NEITHER *MITTRA* NOR *SQUIRE* DISCLOSE A "**PLURALITY** OF GROUP CONTROLLERS"

Squire is not cited in the Final Office Action as disclosing a plurality of group controllers. *Squire* appears to be cited merely as disclosing a load balancer to address the Applicant's argument in response to the first Office Action that *Mittra* failed to disclose a load balancer. Yet the Applicant observes that in discussing network device 500 of Figure 5, *Squire* explains that network device includes a controller 502 that discriminates network traffic for caching and/or load balancing based on network session information identified by network address translator. (Col. 7, line 52 – Col. 8, line 3.) However, the controller 502 of *Squire* has nothing to do with a multicast or broadcast, and therefore controller 502 cannot be considered to be a group controller as featured in Claim 1.

Mittra only discloses "a ***single*** group security controller" and each of Figures 1-3 only depicts a ***single*** GSC. (Col. 6, lines 62-67; emphasis added.) While *Mittra* describes the use of TI servers to manage sub-groups within the group, the TI servers are merely special group members that perform a limited number of functions in place of the single GSC and only for those group members that are below the TI server in the group hierarchy. Thus, the TI

servers are not group controllers since each TI server can only affect group members below it in the group hierarchy.

In contrast to *Mittra*, Claim 1 features "a plurality of group controllers." Furthermore, Claim 1 recites that the plurality of group controllers includes "a first group controller," "a second group controller," and "a third group controller," and Claim 1 features different functions and interactions among the three group controllers and the network nodes service by the group controllers.

Because *Mittra* and *Squire*, either alone or in combination, fail to disclose, teach, suggest, or in any way render obvious "a plurality of group controllers," the Appellants respectfully submit that, for at least the reasons stated above, Claim 1 is allowable over the art of record and is in condition for allowance.

(4) NEITHER *MITTRA* NOR *SQUIRE* DISCLOSE THAT "EACH GROUP CONTROLLER OF THE PLURALITY OF GROUP CONTROLLERS IS A **REPLICA** OF A PARTICULAR GROUP CONTROLLER"

The approach of Claim 1 features that "each group controller of the plurality of group controllers is a ***replica*** of a particular group controller," yet *Mittra* utterly fails to disclose anything that can be interpreted as multiple group controllers that are each a ***replica*** of one of the group controllers. Therefore, the Examiner has failed to establish a prima facie case of obviousness because this feature of Claim 1 is neither taught nor suggested by *Mittra*.

As discussed above, FIG. 10A depicts a set of group controllers, GC, GC1,...,GCN, and the Application explains: "***Each group controller*** GC, GC1, etc., is a ***replica*** of a group controller of the type shown in FIG. 6B and exemplified by group controller 501." (Application, page 37, line 22 – page 38, line 1; emphasis added.) Thus, if one of the group controllers fails, the multicast or broadcast does not fail because there is at least one other group controller that is a replica of the same group controller that can manage the multicast or broadcast. In contrast, in *Mittra*, if the single GSC fails, the multicast fails since there are no other GSCs, little less GSCs that are replicas of the same GSC as the failed GSC.

The portions of *Mittra* relied upon as allegedly disclosing that "each group controller of the plurality of group controllers is a replica of a particular group controller" have been discussed at length above. Not only do those cited portions of *Mittra* fail to disclose a

plurality of group controllers, the cited portions also fail to disclose anything that can possibly be interpreted as disclosing a plurality of group controllers that are each replicas of a particular group controller. While *Mittra* describes the use of TI servers, *Mittra* expressly states that the TI servers as special senders and receivers and further explains that a TI server can only act as proxy for the GSC in performing certain group management functions ***for the sub-group below the TI server***. Therefore, the TI servers are merely members of the group as opposed to being group controllers, and the TI servers are not replicas of anything because each TI server only serves the individual sub-group below the TI server. Therefore, each TI server is unique because each TI server services a different sub-group below it, and there are no descriptions in *Mittra* of multiple TI servers serving the same sub-group.

The Appellants note that while Claim 21 includes a similar feature, namely "a first group controller comprising information that is replicated in a plurality of group controllers," the same portion of *Mittra* is cited in Claim 21 as in Claim 1, and thus the rejection of this similar feature in Claim 21 also fails to disclose anything relating to multiple group controllers that are replicas of a particular group controller.

Squire is not cited in the Final Office Action as disclosing that "each group controller of the plurality of group controllers is a replica of a particular group controller." S*quire* appears to be cited merely as disclosing a load balancer to address the Applicant's argument in response to the first Office Action that *Mittra* failed to disclose a load balancer. As explained above, the controller 502 of *Squire* has nothing to do with a multicast or broadcast, and therefore controller 502 cannot be considered to be a group controller as featured in Claim 1, little less that controller 502 is a replica of a particular group controller.

Because *Mittra* and *Squire*, either alone or in combination, fail to disclose, teach, suggest, or in any way render obvious "each group controller of the plurality of group controllers is a ***replica*** of a particular group controller," the Appellants respectfully submit that, for at least the reasons stated above, Claim 1 is allowable over the art of record and is in condition for allowance.

(5) NEITHER *MITTRA* NOR *SQUIRE* DISCLOSE LOGICAL ARRANGEMENT OF NETWORK NODES AND GROUP CONTROLLERS BASED ON "A **BINARY** TREE"

The approach of Claim 1 involves "a binary tree," and in particular features that "the network nodes and the plurality of group controllers are logically organized in a ***binary tree*** that represents the network nodes and the plurality of group controllers, in which leaf nodes of the binary tree represent network nodes that are joining or leaving the secure multicast or broadcast group, intermediate nodes represent other nodes, and **root nodes** represent the **plurality of group controllers**." (Emphasis added.) Furthermore, Claim 1 features "creating and storing a new group session key for each network node represented ***in each branch of the binary tree*** that is affected by adding or deleting the network node from the secure multicast or broadcast group." (Emphasis added.)

Recall from the discussion above that the reason why a "binary tree" is different than just a "tree" is that each node has two and only two child nodes, as depicted in FIG. 5, FIG. 6B, and FIG. 7B of the Application. FIG. 6B depicts a branch 621 of binary tree 50 that includes the nodes affected by user A joining the multicast, and FIG. 7B depicts a branch 720 that include the nodes affected by user C leaving the multicast.

In *Mittra*, Figures 1-3 depict systems that each includes a ***single*** group security controller (GSC), plus multiple senders (S), receivers (R), TI servers, and multicast/unicast networks. (Col 6, lines 4-5, 20-21, and 39-40.) The elements of the systems of Figures 1-3 are arranged in a hierarchical, tree-like fashion, with the single GSCs as the root node, senders (S) and receivers (R) as the leaf nodes, and the multicast/unicast networks and TI servers as the intermediate nodes. However, the tree-like hierarchy of Figures 1-3 of *Mittra* are ***not binary trees*** because numerous nodes have more than two child nodes. For example, in Figure 1, multicast/unicast network node 112A has six child nodes, S 113A, R 114A, TI 115A, TI 115B, TI 115C, and R 114 B, and both multicast/unicast network nodes 112B and 112D have three child nodes. Similarly, in Figures 2 and 3, multicast/unicast network node 12A has six child nodes and multicast/unicast network nodes 12B and 12D each have three child nodes.

The Final Office Action cites "Fig. 1-3, col. 6 line 4 to line 45" as allegedly disclosing the features of the preamble of Claim 1 that includes the "binary tree" and which types of nodes in the tree represent which types entities involved in the multicast, and the Final Office

Action cites "col. 8 line 45 to line 67" as allegedly disclosing the "branch of the binary tree..." Yet nothing in these cited portions of *Mittra* disclose anything related to a binary tree. Figures 1-3, as explained above, include numerous nodes with more than two child nodes, and therefore, none of Figures 1-3 disclose a binary tree. Column 6, lines 4-45 describes Figures 1-3, but nowhere in that description of Figures 1-3 is there any discussion of anything that could be interpreted as a binary tree. In fact, an electronic search of *Mittra* reveals that there are no occurrences of the word "binary."

Finally, Column 8, lines 45-67 describes changing the group key (Kgrp) when a member leaves the group, along with several group key distribution schemes, including the use of individual secure channels between the GSC and group members or the use of a single multicast with multiple copies of the group key encrypted with each member's secure channel key.

None of the cited portions of *Mittra* disclose anything that could be possibly interpreted as a binary tree or that would suggest a binary tree, and none of the cited portions of *Mittra* discloses anything that could be possibly interpreted as either disclosing or suggesting the creating and storing of a new group session key for network nodes represented in a branch of the binary tree affected by adding or deleting a network node from the group, as featured in Claim 1. Furthermore, in each of Figures 1-3 in *Mittra*, the root node corresponds to one and only one GSC, whereas Claim 1 features that the root nodes represent the plurality of group controllers.

During the Examiner Interview, the Appellants presented these arguments, and the Examiner replied that the trees depicted in Figures 1-3 in *Mittra* could be traversed using only two child nodes at each parent node, and therefore the Examiner concluded that as a result of that interpretation, Figures 1-3 could be considered as disclosing a binary tree as featured in Claim 1. However, the Examiner's interpretation that the trees of Figures 1-3 can be traversed based on only two child nodes of each parent node that includes more than two child nodes is completely illogical because such an interpretation would effectively ignore any additional child nodes of those parent nodes with more than two child nodes.

For example, if in Figure 1, multicast/unicast network 112A were to be traversed based on only two child nodes, which two child nodes would be selected and which four child nodes would be ignored? If the two selected child nodes were TI 115A and TI 115C,

then S 113A, R 114A, and R 114B would be ignored, in addition to TI 115B and all nodes below TI 115B, which includes three more multicast/unicast networks, two more TI servers, and four more receivers. However, if the two selected child nodes were R 114A and R 114B, then almost all of the remaining nodes of Figure 1 would be ignored. If Figures 1-3 of *Mittra* did depict binary trees, then there would need to be considerably more intermediate nodes between the first multicast/unicast networks just below the GSCs, which is clearly not the case.

Squire is not cited in the Final Office Action as disclosing a "binary tree," and the Appellant has not found any tree-like structure in *Squire*, little less a binary tree.

Because *Mittra* and *Squire*, either alone or in combination, fail to disclose, teach, suggest, or in any way render obvious "a ***binary* tree** that represents the network nodes and the plurality of group controllers, in which leaf nodes of the binary tree represent network nodes that are joining or leaving the secure multicast or broadcast group, intermediate nodes represent other nodes, and **root nodes** represent the **plurality of group controllers**," the Appellants respectfully submit that, for at least the reasons stated above, Claim 1 is allowable over the art of record and is in condition for allowance.

(6) NEITHER *MITTRA* NOR *SQUIRE* DISCLOSE "RECEIVING A **REQUEST** TO ADD OR DELETE A NETWORK NODE...**FROM A LOAD BALANCER**"

The approach of Claim 1 includes "***receiving a* request *to add or delete a network node*** of the secure multicast or broadcast group from a **load balancer** that is coupled to the plurality of group controllers." (Emphasis added.) *Mittra* discloses nothing about a load balancer, little less that a request to add or delete a network node from the group is received from the load balancer. The Appellant was successful in convincing the Examiner of this in the response to the first Office Action, because the Final Office Action states that "Mittra does not specifically disclose [a] load balancer in [a] network environment to man[a]ge the network traffic."

The Final Office Action relies upon *Squire* as allegedly disclosing a "computing environment where load balancing devices have been added to networks components in an effort to more efficiently manage the finite bandwidth of the network capacity, and a load balancing device is designed to route network traffic through optimal data paths based on a

number of traffic-centric and network-centric parameters, in accordance with a network management strategy (col. 7 line 53 to col. 8 line 4)." As a preliminary matter, the cited portion of *Squire* fails to disclose the material recited in the Office Action. Instead, the passage above from the Final Office Action from "a load balancing device..." to "in accordance with a network management strategy" is from Column 2, lines 39-43 of *Squire*. However, even the correct citation from *Squire* fails to disclose anything about receiving a request to add or delete a network node from a load balancer.

Therefore, while *Squire* discusses a new type of network device that integrates the functions of a load balancer and a network address translator, there is nothing in *Squire* about multicasts or broadcasts, little less the receiving of a request from a load balancer to add or delete a member of a multicast.

Also, the cited portion of *Squire* merely describes a network device 500 that can be implemented as an integrated network address translator (NAT)/load balancer or an integrated NAT/network cache/load balancer. Nothing in the cited portion of *Squire* discloses anything about receiving a request from a load balancer to add or delete a network node from a multicast or broadcast group. Rather, the purpose of network device 500 is to integrate "the feature rich servers of NAT 501 with a host of network caching/load balancing features to provide a network cache/load balancing network device 500 that is OSI layer 5 aware." (Col. 7, lines 47-51.) Making a network device that performs load balancing based on layer 5 information discloses nothing about the incorporation of a load balancer with a plurality of group controllers, little less receiving a request from the load balancer to change the membership of the multicast group. Therefore, the Examiner has failed to establish a prima facie case of obviousness because this feature of Claim 1 is neither taught nor suggested by *Mittra*.

Furthermore, the Final Office Action alleges that it "would have been obvious to one of ordinary skill in the art at the time of the invention to combine the teachings of Mittra and Squire, because Squire's optimized method to discriminate network traffic based on network session information for selecting traffic to determine network routing, would not only analyzes network traffic to identify network session information and based on the network session information network performing load balancing in accordance with a predetermined network management strategy, but will also provide an integrated network load balancer,

wherein controller load balancing device is implemented as a front end for a collection of servers."

However, this alleged motivation to combine the teachings of *Mittra* and *Squire* are merely a collection of quotations from the background section of *Squire* that describes the load balancing and the deficiencies of prior load balancing devices. For example, in the background section, *Squire* states: "None of the prior art network caches or load balancing devices, however, are **optimized to discriminate network traffic based on network session information**, e.g., at OSI layer 5, as means **for selecting traffic to cache or determining network routing**." (Col. 2, lines 53-57; emphasis added to highlight the same wording in the Final Office Action.) Also, *Squire* explains: "a network device 104a-104n integrating NAT features with a network cache/load balancer **analyzes network traffic to identify network session information and based**, at least in part, **on the network session information network** caching/**load balancing is performed in accordance with a predetermined network management strategy**. (Col. 6, line 63 – Col. 7, line 2; emphasis added to highlight the same wording in the Final Office Action.) Finally, *Squire* states in the Background section: "Thus, a load balancing **device is** typically **implemented as a front end for a collection of servers**." (Col. 2, lines 45-47; emphasis added to highlight the same wording in the Final Office Action.)

It is unclear to the Appellants how stitching together three disparate passages from *Squire*, in which one describes a problem with prior load balancing devices, another an advantage of *Squire's* approach, and the third the function of a load balancing device, represents a proper teaching, suggestion, or motivation to combine the integrated NAT/load balancing device of *Squire* with the group security controller approach of *Mittra*. As discussed above at length, *Mittra* fails to disclose multiple group controllers, hence there is nothing in *Mittra* to load balance. And *Squire* says nothing about a "multicast" or a "broadcast," and an electronic search of *Squire* shows now occurrences of either word. Thus, it appears to the Appellants that the only motivation to combine *Mittra* with *Squire* is impermissible hindsight reasoning by the Examiner, with the only justification being a mish-mash of quotations from *Squire* that have no relation to each other, little less any relation to why one of ordinary skill in the art at the time the invention was made would combine the teachings of *Mittra* and *Squire*.

Furthermore, even if combined, *Mittra* and *Squire* do not disclose numerous features of Claim 1, and therefore the Examiner has failed to establish a prima facie case of obviousness because this feature of Claim 1 is neither taught nor suggested by *Mittra* or *Squire*, either alone or in combination.

Because *Mittra* and *Squire*, either alone or in combination, fail to disclose, teach, suggest, or in any way render obvious "***receiving a* request *to add or delete a network node*** of the secure multicast or broadcast group from a **load balancer** that is coupled to the plurality of group controllers," the Appellants respectfully submit that, for at least the reasons stated above, Claim 1 is allowable over the art of record and is in condition for allowance.

(7) CONCLUSION OF DISCUSSION OF CLAIM 1 AND *MITTRA & SQUIRE*

The approach for multicasting in *Mittra* and the approach of Claim 1 are fundamentally different in that *Mittra* using only a "single" group security controller (GSC), whereas Claim 1 features "a **plurality** of group controllers" and expressly recites the roles of three group controllers of the plurality of group controllers. While *Mittra* describes the use of trusted intermediary (TI) servers, *Mittra* expressly states that the TI servers are only special group members and that each TI server only performs specified functions of the single GSC for the corresponding sub-group of the TI server. Therefore, the only group controller in *Mittra* is the single GSC, whereas Claim 1 features "a **plurality** of group controllers." Therefore, the Examiner has failed to establish a prima facie case of obviousness because "a **plurality** of group controllers" is neither taught nor suggested by *Mittra*.

Furthermore, Claim 1 features that each group controller of the plurality of group controllers is a **replica** of a particular group controller, whereas *Mittra's* single GSC is not a replica of anything, nor are any of the TI servers in *Mittra* replicas of each other or of the GSC since each TI server only manages the particular sub-group below that TI server. Therefore, the Examiner has failed to establish a prima facie case of obviousness because "each group controller of the plurality of group controllers is a **replica** of a particular group controller" is neither taught nor suggested by *Mittra*.

In addition, Claim 1 features a ***binary* tree** and recites how the network nodes and group controllers are logically represented in the binary tree wherein the **root nodes** represent the plurality of group controllers and the use of a particular branch of the binary tree. In

contrast, *Mittra* only discloses a general hierarchical arrangement or tree that includes many non-binary features, and thus *Mittra* fails to disclose a binary tree with the organization featured in Claim 1 or the use of a branch of the binary tree in distributing a new group session key. Therefore, the Examiner has failed to establish a prima facie case of obviousness because neither "a binary tree" nor "root nodes [that] represent the plurality of group controllers" is taught or suggested by *Mittra*.

Finally, Claim 1 features receiving from a **load balancer** a request to add or delete a network node from the group, yet *Mittra* discloses nothing about load balancers or load balancing and *Squire* only discloses an integrated network address translator/load balancer, which has nothing to do with multicasts or broadcasts. Therefore, the Examiner has failed to establish a prima facie case of obviousness because "receiving from a **load balancer** a request to add or delete a network node form the group" is neither taught nor suggested by either *Mittra* or *Squire*.

Because *Mittra* and *Squire*, either alone or in combination, fail to disclose, teach, suggest, or in any way render obvious "a ***plurality* of group controllers**," "wherein **each group controller** of the plurality of group controllers **is a *replica*** of a particular group controller," "wherein the network nodes and the plurality of group controllers are **logically organized in a *binary* tree** that represents the network nodes and the plurality of group controllers...and **root nodes** represent the plurality of group controllers," and "**receiving a request** to **add or delete a network node** of the secure multicast or broadcast group **from a *load balancer*** that is coupled to the plurality of group controllers," the Appellants respectfully submit that, for at least the reasons stated above, Claim 1 is allowable over the art of record and is in condition for allowance.

C. Claims 11, 31, and 41 Are Patentable Over *Mittra* in View of *Squire*

Claims 11, 31, and 41 contain features that are similar to those described above with respect to Claim 1, and in particular all feature "a ***plurality* of group controllers**," "wherein **each group controller** of the plurality of group controllers **is a *replica*** of a particular group controller," "wherein the network nodes and the plurality of group controllers are **logically organized in a *binary* tree** that represents the network nodes and the plurality of group controllers...and **root nodes** represent the plurality of group controllers," and "**receiving a**

request to **add or delete a network node** of the secure multicast or broadcast group **from a *load balancer*** that is coupled to the plurality of group controllers." Therefore, based on at least the reasons stated above with respect to Claim 1, the Applicant respectfully submits that Claims 11, 31, and 41 are allowable over the art of record and is in condition for allowance.

D. Claims 2, 12, 32, and 42 Are Patentable Over *Mittra* in View of *Squire*

Claims 2, 12, 32, and 42 are dependent upon Claims 1, 11, 31, and 41, respectively, and thus include each and every feature of the corresponding independent claims. Each of Claims 2, 12, 32, and 42 is therefore allowable for the reasons given above for the Claims 1, 11, 31, and 41. Therefore, it is respectfully submitted that Claims 2, 12, 32, and 42 are allowable for the reasons given above with respect to Claims 1, 11, 31, and 41.

In addition, each of Claims 2, 12, 32, and 42 introduces one or more additional limitations that independently render it patentable. The Final Office Action states that "the system of Mittra and Squire teaches and describes distributing a group session key further comprises: ***receiving a*** **token value** ***at the third group controller*** to ***designate the third group controller as having*** **permission** to selectively generate the group session key and to generate node keys associated with the intermediate nodes and the leaf nodes" and "creating and storing the group session key **only when** ***the third group controller has the*** **token value** (col. 2 line 8 to line 20)." (Emphasis added.) Because the Final Office Action fails to state from which reference the citation is taken (or even what in the reference corresponds to the "token value" featured in Claims 2, 12, 32, and 42), both references will be addresses herein.

In *Mittra*, the cited portion describes the central problem of controlling data transmitted in a multicast as the senders and receivers sharing the group key and managing the group key as the group membership changes in a secure multicast (e.g., when members join or leave the group). (Col. 2, lines 8-20.) But nothing in this portion of *Mittra*, nor any other portion of *Mittra*, discloses anything about "a token value" that is used by one group controller of the plurality of group controllers to designate when that group controller has permission to generate and store the group session key, as featured in Claims 2, 12, 32, and 42. Rather, *Mittra* describes the function of controlling group membership and generating keys for all members of the group as being the responsibility only of the single GSC. (Col. 7, lines 30-32; Col. 8, lines 17-28; Col. 8, lines 51-52.) *Mittra's* description that group keys must be managed when group members join and leave the group, while being true

for all multicast groups, says nothing about the use of a token value. In contrast to *Mittra*, Claims 2, 12, 32, and 42 feature how one particular group controller from multiple group controllers is designated as having the responsibility for creating new group session keys.

In *Squire*, the cited portion merely describes the routing of packets at the network layer, such as OSI layer 3 as used in the Internet Protocol (IP) for the Internet and that prior art network caches discriminate network traffic based on the transport layer, such as OSI layer 4 information. (Col. 2, lines 8-20.) The routing of packets on the Internet at network layer 3 and network caches using transport layer 4 says nothing about "a token value" that is used by one group controller of the plurality of group controllers to designate when that group controller has permission to generate and store the group session key, as featured in Claims 2, 12, 32, and 42.

Because *Mittra* and *Squire*, either alone or in combination, fail to disclose, teach, suggest, or in any way render obvious "***receiving a* token value *at the third group controller*** to ***designate the third group controller as having* permission** to selectively generate the group session key and to generate node keys associated with the intermediate nodes and the leaf nodes" and "creating and storing the group session key **only when *the third group controller has the* token value,**" the Appellants respectfully submit that, for at least the reasons stated above, Claims 2, 12, 32, and 41 are allowable over the art of record and is in condition for allowance.

E. Claims 3-10, 13-20, 33-40, and 43-50 Are Patentable Over *Mittra* in View of *Squire*

Claims 3-10, 13-20, 33-40, and 43-50 are dependent upon Claims 1, 11, 31, and 41, respectively, and thus include each and every feature of the corresponding independent claims. Each of Claims 3-10, 13-20, 33-40, and 43-50 is therefore allowable for the reasons given above for the Claims 1, 11, 31, and 41. In addition, each of Claims 3-10, 13-20, 33-40, and 43-50 introduces one or more additional limitations that independently render it patentable. However, due to the fundamental differences already identified, to expedite the positive resolution of this case a separate discussion of those limitations is not included at this time. Therefore, it is respectfully submitted that Claims 3-10, 13-20, 33-40, and 43-50 are allowable for the reasons given above with respect to Claims 1, 11, 31 and 41.

F. Claims 21, 51, 61, and 71 Are Patentable Over *Mittra* in View of *Squire*

Claims 21, 51, 61, and 71 contain features that are similar to those described above with respect to Claim 1, and in particular all feature "a ***plurality of group controllers***," "a first group controller comprising information that is ***replicated*** in a plurality of group controllers," "wherein the network nodes and the plurality of group controllers are logically organized in a **binary tree** that represents the network nodes and the plurality of group controllers...and **root nodes** that represent the plurality of group controllers," and "receiving a request to add or delete a network node from a load balancer that controls distribution of requests to the plurality of group controllers." Therefore, based on at least the reasons stated above with respect to Claim 1, the Applicant respectfully submits that Claims 21, 51, 61, and 71 are allowable over the art of record and is in condition for allowance.

G. Claims 22, 52, 62, and 72 Are Patentable Over *Mittra* in View of *Squire*

Claims 22, 52, 62, and 72 are dependent upon Claims 21, 51, 61, and 71, respectively, and thus include each and every feature of the corresponding independent claims. Each of Claims 22, 52, 62, and 72 is therefore allowable for the reasons given above for the Claims 21, 51, 61, and 71. Therefore, it is respectfully submitted that Claims 22, 52, 62, and 72 are allowable for the reasons given above with respect to Claims 21, 51, 61, and 71.

In addition, each of Claims 22, 52, 62, and 72 introduces one or more additional limitations that independently render it patentable. The Final Office Action states that "the system of Mittra and Squire teaches and describes...the steps of generating the group session key **only when** the ***first group controller is designated as a*** **master group controller** that is authorized to join nodes and generate group session keys (col. 12 line 50 to col. 13 line 18)." (Emphasis added.) While the Final Office Action fails to state from which reference the citation is taken, only *Mittra* is addressed herein because it appears to the Appellants that that is the reference being cited as the disclosure of *Squire* ends as Col. 12, line 61 and thus does not include any disclosure through Column 13, line 18 as cited in the Final Office Action.

In *Mittra*, the cited portion describes the functioning of the TI servers, explaining that the TI servers are grouped according to "levels," that the TI servers pass on messages from their parents to their children, and that the TI servers manage the group membership of their sub-group under them. (Col. 12, line 50 – Col. 13, line 18.) However, there is nothing in this cited portion of *Mittra* that describes a condition placed on the TI servers performing these

functions, such as that each TI server only generates a group key for its subgroup when the TI server is somehow designated as a master group controller. Furthermore, none of the TI servers can generate a group session key for all members of the group; rather, each TI server can only generate session keys for members below that TI server.

In contrast to *Mittra*, Claims 22, 52, 62, and 72 feature a specific condition under which the first group controller generates the group session key, namely that the first group controller is designated as a master group controller that is authorized to join nodes and generate group session keys, which applies to the entire group, not just a sub-group.

Because *Mittra* and *Squire*, either alone or in combination, fail to disclose, teach, suggest, or in any way render obvious "generating the group session key **only when** the ***first group controller is designated as a*** **master group controller** that is authorized to join nodes and generate group session keys," the Appellants respectfully submit that, for at least the reasons stated above, Claims 2, 12, 32, and 41 are allowable over the art of record and is in condition for allowance.

H. Claims 23, 53, 63, and 73 Are Patentable Over *Mittra* in View of *Squire*

Claims 23, 53, 63, and 73 are dependent upon Claims 21, 51, 61, and 71, respectively, and thus include each and every feature of the corresponding independent claims. Each of Claims 23, 53, 63, and 73 is therefore allowable for the reasons given above for the Claims 21, 51, 61, and 71. Therefore, it is respectfully submitted that Claims 23, 53, 63, and 73 are allowable for the reasons given above with respect to Claims 21, 51, 61, and 71.

In addition, each of Claims 23, 53, 63, and 73 introduces one or more additional limitations that independently render it patentable. The Final Office Action states that "the system of Mittra and Squire teaches and describes...the steps of ***successively designating*** different ones of the group controllers ***as the*** **master** ***group controller*** in real time (col. 12 line 50 to col. 18 line 18.) (Emphasis added.) While the Final Office Action fails to state from which reference the citation is taken, only *Mittra* is addressed herein because it appears to the Appellants that that is the reference being cited as the disclosure of *Squire* ends as Col. 12, line 61 and thus does not include any disclosure through Column 13, line 18 as cited in the Final Office Action.

In *Mittra*, the cited portion describes the functioning of the TI servers, explaining that the TI servers are grouped according to "levels," that the TI servers pass on messages from

their parents to their children, and that the TI servers manage the group membership of their sub-group under them. (Col. 12, line 50 – Col. 13, line 18.) However, there is nothing in this cited portion of *Mittra* that describes successively designating the TI servers as a master TI server that is then responsible for performing these functions when that designation is made, such as that each TI server only generates a group key for its subgroup when the TI server is somehow designated as a master TI server. Furthermore, none of the TI servers can generate a group session key for all members of the group; rather, each TI server can only generate session keys for members below that TI server.

In contrast to *Mittra*, Claims 23, 53, 63, and 73, which are dependent upon Claims 22, 52, 62, and 72, feature a specific manner for designating which group controller of the plurality of group controllers is the master and therefore through such designation has permission to generate the group session key for the group. This means that each group controller of the plurality of group controllers has the ability to be the master and perform the functions permitted by that designation, whereas in *Mittra*, each TI server always performs the specified functions for the TI server's subgroup and there is never a decision about which TI server has permission to do so and thus no need for a token to designate a master TI server among the TI servers.

Because *Mittra* and *Squire*, either alone or in combination, fail to disclose, teach, suggest, or in any way render obvious "***successively designating*** different ones of the group controllers ***as the* master *group controller*** in real time," the Appellants respectfully submit that, for at least the reasons stated above, Claims 2, 12, 32, and 41 are allowable over the art of record and is in condition for allowance.

I. Claims 24, 54, 64, and 74 Are Patentable Over *Mittra* in View of *Squire*

Claims 24, 54, 64, and 74 contain features that are similar to those described above with respect to Claim 1, and in particular all feature "a ***plurality* of group controllers**," "a logical arrangement of the network nodes in **a *binary* tree** structure" in which "the plurality of group controllers correspond to the **root node**," and "**load balancing** traffic emanating from a plurality of network nodes to the plurality of group controllers." Therefore, based on at least the reasons stated above with respect to Claim 1, the Applicant respectfully submits that Claims 24, 54, 64, and 74 are allowable over the art of record and is in condition for allowance.

J. Claims 25, 55, 65, and 75 Are Patentable Over *Mittra* in View of *Squire*

Claims 25, 55, 65, and 75 are dependent upon Claims 24, 54, 64, and 74, respectively, and thus include each and every feature of the corresponding independent claims. Each of Claims 25, 55, 65, and 75 is therefore allowable for the reasons given above for the Claims 24, 54, 64, and 74. Therefore, it is respectfully submitted that Claims 25, 55, 65, and 75 are allowable for the reasons given above with respect to Claims 24, 54, 64, and 74.

In addition, each of Claims 25, 55, 65, and 75 introduces one or more additional limitations that independently render it patentable. The Final Office Action states that "the system of Mittra and Squire teaches and describes [the] step of distributing further comprises: ***circulating a*** **token** ***among the plurality of group controllers*** to ***designate the one group controller as having*** **permission** to selectively generate the group session key and keys associated with the intermediate nodes and the leaf nodes" and "**selectively** ***generating*** the group session key based upon the circulating step (col. 2 line 8 to line 20, and col. 13 line 39 to line 56" (Emphasis added.) While the Final Office Action fails to state from which reference the citation is taken, only *Mittra* is addressed herein because it appears to the Appellants that that is the reference being cited as the disclosure of *Squire* ends as Col. 12, line 61 and thus does not include any disclosure from a Column 13 as cited in the Final Office Action.

In *Mittra*, the first cited portion describes the central problem of controlling data transmitted in a multicast as the senders and receivers sharing the group key and managing the group key as the group membership changes in a secure multicast (e.g., when members join or leave the group). (Col. 2, lines 8-20.) But nothing in this portion of *Mittra*, nor any other portion of *Mittra*, discloses anything about "a token" that is circulated "among the plurality of group controllers to designate the one group controller as having permission to selectively generate the group session key and keys...," as featured in Claims 25, 55, 65, and 75. Rather, *Mittra* describes the function of controlling group membership and generating keys for all members of the group as being the responsibility only of the single GSC. (Col. 7, lines 30-32; Col. 8, lines 17-28; Col. 8, lines 51-52.) *Mittra's* description that group keys must be managed when group members join and leave the group is true for all multicast groups, whereas Claims 25, 55, 65, and 75 features how one particular group

controller from multiple group controllers is designated as having the responsibility for generating new group session keys.

The second cited portion of *Mittra* describes the joining of a member to the secure multicast group using a trusted intermediary (TI) based on the access point of the member in the hierarchy of the group, such that the joining member either the GSC or the appropriate TI server depending on where in the hierarchy of Figure 1 the member is located. (Col. 13, lines 39-56.) However, there is nothing in this portion of *Mittra* about a token being circulated among multiple group controllers, little less that the token designates which group controller among the group controller has permission to generate the group session key, as featured in Claim 1. Rather, in *Mittra*, the entity responsible for handling the joining member and generating a group session key for the sub-group that the joining member will belong to is always the same because that responsibility is based on the joining member's location in the hierarchy, and *Mittra* discloses nothing about the joining member's location changing.

Furthermore, in *Mittra*, the group session key that is generated when the member joins is only for the subgroup that the member belongs to and not for the entire multicast, not for the group as a whole. And there is nothing in *Mittra* that suggests that the generation of a group session key is selectively performed based on circulating the token, as featured in Claims 25, 55, 65, and 75.

Because *Mittra* and *Squire*, either alone or in combination, fail to disclose, teach, suggest, or in any way render obvious "***circulating a* token *among the plurality of group controllers*** to ***designate the one group controller as having* permission** to selectively generate the group session key and keys associated with the intermediate nodes and the leaf nodes" and "**selectively *generating*** the group session key based upon the circulating step," the Appellants respectfully submit that, for at least the reasons stated above, Claims 25, 55, 65, and 75 are allowable over the art of record and is in condition for allowance.

K. Claims 26-30, 56-60, 66-70, and 76-80 Are Patentable Over *Mittra* in View of *Squire*

Claims 26-30, 56-60, 66-70, and 76-80 are dependent upon Claims 24, 54, 64, and 74, respectively, and thus include each and every feature of the corresponding independent claims. Each of Claims 26-30, 56-60, 66-70, and 76-80 is therefore allowable for the reasons given above for the Claims 24, 54, 64, and 74. In addition, each of Claims 26-30, 56-60,

66-70, and 76-80 introduces one or more additional limitations that independently render it patentable. However, due to the fundamental differences already identified, to expedite the positive resolution of this case a separate discussion of those limitations is not included at this time. Therefore, it is respectfully submitted that Claims 26-30, 56-60, 66-70, and 76-80 are allowable for the reasons given above with respect to Claims 24, 54, 64, and 74.

L. Conclusion and Prayer for Relief

Based on the foregoing, the Appellants respectfully submits that the rejections of Claims 1-80 under 35 U.S.C. § 103(a) as allegedly unpatentable over *Mittra* in view of *Squire* lacks the requisite factual and legal bases. The Appellants respectfully submit that the imposed rejections under 35 U.S.C. § 103(a) over *Mittra* in view of *Squire* are **not** viable and respectfully solicit the Honorable Board to **reverse** each of the imposed rejections under 35 U.S.C. § 103(a).

To the extent necessary to make this reply timely filed, the Applicant petitions for an extension of time under 37 C.F.R. § 1.136.

If any applicable fee is missing or insufficient, throughout the pendency of this application, the Commissioner is hereby authorized to any applicable fees and to credit any overpayments to our Deposit Account No. 50-1302.

Respectfully submitted,
HICKMAN PALERMO TRUONG & BECKER LLP

Craig G. Holmes
Reg. No. 44,770

Date: June 22, 2005

2055 Gateway Place, Suite 550
San Jose, CA 95110-1089
Telephone: (408) 414-1207
Facsimile: (408) 414-1076

Claims Appendix

CERTIFICATE OF MAILING

I hereby certify that this correspondence is being deposited with the United States Postal Service as first class mail in an envelope addressed to: Board of Patent Appeals and Interferences, United States Patent and Trademark Office, P.O. Box 1450, Alexandria, VA 22313-1450.

on 6/22/05 by Tracy Reynolds

VIII. CLAIMS APPENDIX

1. (Previously Presented) A method for managing addition and deletion of network nodes from and to a secure multicast or broadcast group of network nodes in a communications network without a single point of failure, wherein each of the network nodes is associated with one of a plurality of group controllers, wherein each group controller of the plurality of group controllers is a replica of a particular group controller, and wherein the network nodes and the plurality of group controllers are logically organized in a binary tree that represents the network nodes and the plurality of group controllers, in which leaf nodes of the binary tree represent network nodes that are joining or leaving the secure multicast or broadcast group, intermediate nodes represent other network nodes, and root nodes represent the plurality of group controllers, the method comprising the steps of:

joining a first group controller to the plurality of group controllers in a local network;

establishing a secure communication channel between the first group controller and a second group controller of the plurality of group controllers using a key exchange protocol;

receiving a request to add or delete a network node of the secure multicast or broadcast group from a load balancer that is coupled to the plurality of group controllers;

creating and storing a new group session key for each network node represented in each branch of the binary tree that is affected by adding or deleting the network node from the secure multicast or broadcast group; and

distributing a group session key from a third group controller of the plurality of group controllers to the network nodes.

2. (Previously Presented) A method as recited in Claim 1, wherein distributing a group session key further comprises:

receiving a token value at the third group controller to designate the third group controller as having permission to selectively generate the group session key and to generate node keys associated with the intermediate nodes and the leaf nodes; and

creating and storing the group session key only when the third group controller has the token value.

3. (Previously Presented) A method as recited in Claim 1, wherein distributing a group session key further comprises:
determining whether the secure multicast or broadcast group has a network node that is leaving the secure multicast or broadcast group;
determining which of the intermediate nodes are affected by the leaving node;
updating keys associated with the affected intermediate nodes;
generating a new group session key; and
sending the new group session key to the leaf nodes.

4. (Previously Presented) A method as recited in Claim 3, wherein updating keys comprises:
generating a new key of a parent node of the leaving node; and
encrypting the new key of the parent node with a key of a network node adjacent to the parent node.

5. (Previously Presented) A method as recited in Claim 1, wherein distributing a group session key further comprises:
receiving a request message from one of the network nodes to join the secure multicast or broadcast group;
determining which of the intermediate nodes are affected by the joining node;
updating keys associated with the affected intermediate nodes;
generating a new group session key and a private key of the joining node; and
sending a message comprising the new group session key, the private key, and the updated keys of affected intermediate nodes to the joining node.

6. (Original) A method as recited in Claim 5, wherein updating keys comprises performing a one way hash function on the keys associated with the affected intermediate nodes.

7. (Original) A method as recited in Claim 1, wherein receiving a request comprises receiving the request at a load balancer having a single virtual address that represents the plurality of group controllers.

8. (Previously Presented) A method as recited in Claim 7, further comprising the step of load balancing network traffic that is directed from a plurality of the network nodes to the plurality of group controllers.

9. (Previously Presented) A method as recited in Claim 1, wherein establishing a secure communication channel comprises exchanging a public key of the first group controller with all other group controllers in the plurality of group controllers based upon optimized broadcast Diffie-Hellman protocol.

10. (Previously Presented) A method as recited in Claim 5, wherein establishing a secure communication channel comprises:
receiving a public key value that is broadcast by the joining node;
sending a collective public key value from the network nodes to the joining node;
computing a shared secret key; and
creating and storing a group shared secret key by exchanging private key values.

11. (Previously Presented) A computer-readable medium comprising one or more sequences of instructions for managing addition and deletion of network nodes from and to a secure multicast or broadcast group of network nodes in a communications network without a single point of failure, wherein each of the network nodes is associated with one of a plurality of group controllers, wherein each group controller of the plurality of group controllers is a replica of a particular group controller, and wherein the network nodes and the plurality of group controllers are logically organized in a binary tree that represents the network nodes and the plurality of group controllers, in which leaf nodes of the binary tree represent network nodes that are joining or leaving the secure multicast or broadcast group, intermediate nodes represent other network nodes, and root nodes represent the plurality of group controllers, and which instructions, when executed by one or more processors, cause the processors to carry out the steps of:

joining a first group controller to the plurality of group controllers in a local network;
establishing a secure communication channel between the first group controller and a second group controller of the plurality of group controllers using a public key exchange protocol;
receiving a request to add or delete a network node of the secure multicast or broadcast group from a load balancer that is coupled to the plurality of group controllers;
creating and storing a new group session key for each network node represented in each branch of the binary tree that is affected by adding or deleting the network node from the secure multicast or broadcast group; and
distributing a group session key from a third group controller of the plurality of group controllers to the network nodes.

12. (Previously Presented) A computer-readable medium as recited in Claim 11, wherein distributing a group session key further comprises:
receiving a token value at the third group controller to designate the third group controller as having permission to selectively generate the group session key and to generate node keys associated with the intermediate nodes and the leaf nodes; and
creating and storing the group session key only when the third group controller has the token value.

13. (Previously Presented) A computer-readable medium as recited in Claim 11, wherein distributing a group session key further comprises:
determining whether the secure multicast or broadcast group has a network node that is leaving the secure multicast or broadcast group;
determining which of the intermediate nodes are affected by the leaving node;
updating keys associated with the affected intermediate nodes;
generating a new group session key; and
sending the new group session key to the leaf nodes.

14. (Previously Presented) A computer-readable medium as recited in Claim 3, wherein updating keys comprises:

generating a new key of a parent node of the leaving node; and

encrypting the new key of the parent node with a key of a network node adjacent to the parent node.

15. (Previously Presented) A computer-readable medium as recited in Claim 11, wherein distributing a group session key further comprises:

receiving a request message from one of the network nodes to join the secure multicast or broadcast group;

determining which of the intermediate nodes are affected by the joining node;

updating keys associated with the affected intermediate nodes;

generating a new group session key and a private key of the joining node; and

sending a message comprising the new group session key, the private key, and the updated keys of affected intermediate nodes to the joining node.

16. (Original) A computer-readable medium as recited in Claim 15, wherein updating keys comprises performing a one way hash function on the keys associated with the affected intermediate nodes.

17. (Original) A computer-readable medium as recited in Claim 11, wherein receiving a request comprises receiving the request at a load balancer having a single virtual address that represents the plurality of group controllers.

18. (Previously Presented) A computer-readable medium as recited in Claim 17, further comprising instructions which, when executed by the one or more processors, cause the processors to carry out the step of load balancing network traffic that is directed from a plurality of the network nodes to the plurality of group controllers.

19. (Previously Presented) A computer-readable medium as recited in Claim 11, wherein establishing a secure communication channel comprises exchanging a public key of the first group controller with all other group controllers in the plurality of group controllers based upon Diffie-Hellman protocol.

20. (Previously Presented) A computer-readable medium as recited in Claim 15, wherein establishing a secure communication channel comprises:
receiving a public key value that is broadcast by the joining node;
sending a collective public key value from the network nodes to the joining node;
computing a shared secret key; and
creating and storing a group shared secret key by exchanging private key values.

21. (Previously Presented) A method of managing addition and deletion of network nodes from and to a secure multicast or broadcast group of network nodes in a communications network, wherein each of the network nodes is associated with a first group controller comprising information that is replicated in a plurality of group controllers, and wherein the network nodes and the plurality of group controllers are logically organized in a binary tree that represents the network nodes and the plurality of group controllers, in which leaf nodes of the binary tree represent network nodes that are joining or leaving the secure multicast or broadcast group, intermediate nodes represent other network nodes, and root nodes represent the plurality of group controllers, the method comprising the steps of:
joining the first group controller in a local network in which the plurality of group controllers are coupled;
establishing a secure channel between the first group controller and the plurality of group controllers through secure key exchange;
receiving a request to add or delete a network node from a load balancer that controls distribution of requests to the plurality of group controllers;
generating a new group session key for each network node represented in each branch of the binary tree that is affected by adding or deleting the network node from the secure multicast or broadcast group; and
distributing the group session key from the first group controller to the other group controllers of the plurality of group controllers over the secure channel.

22. (Previously Presented) A method as recited in Claim 21, further comprising the step of generating the group session key only when the first group controller is designated as a master group controller that is authorized to join network nodes and generate group session keys.

23. (Previously Presented) A method as recited in Claim 22, further comprising the step of successively designating different group controllers of the plurality of group controllers as the master group controller in real time.

24. (Previously Presented) A method for creating a secure multicast or broadcast group, the method comprising the steps of:

establishing a secure communication channel among a plurality of group controllers via a public key exchange protocol;

load balancing traffic emanating from a plurality of network nodes to the plurality of group controllers; and

distributing a group session key by one of the group controllers based upon a logical arrangement of the network nodes in a binary tree structure, the binary tree structure having a root node, intermediate nodes, and leaf nodes, wherein the plurality of network nodes correspond to leaf nodes of the binary tree structure and the plurality of group controllers correspond to the root node.

25. (Original) The method as recited in Claim 24, wherein the step of distributing further comprises:

circulating a token among the plurality of group controllers to designate the one group controller as having permission to selectively generate the group session key and keys associated with the intermediate nodes and the leaf nodes; and

selectively generating the group session key based upon the circulating step.

26. (Previously Presented) The method as recited in Claim 24, wherein the step of distributing further comprises:

detecting whether the secure multicast or broadcast group has a network node that is leaving the secure multicast or broadcast group;

determining which of the intermediate nodes are affected in response to the detecting step;

updating keys associated with the affected intermediate nodes;
generating a new group session key; and
sending the new group session key to the leaf nodes.

27. (Previously Presented) The method as recited in Claim 26, wherein the step of updating comprises:
generating a new key of a parent node of the leaving node; and
encrypting the new key of the parent node with a key of a network node adjacent to the parent node.

28. (Previously Presented) The method as recited in Claim 24, wherein the step of distributing further comprises:
receiving a request message from one of the plurality of network nodes to join the secure multicast or broadcast group;
determining which of the intermediate nodes are affected in response to the receiving step;
updating keys associated with the affected intermediate nodes;
generating a new group session key and a private key of the joining node; and
sending a message comprising the new group session key, the private key, and the updated keys of affected intermediate nodes to the joining node.

29. (Original) The method as recited in Claim 28, wherein the step of updating comprises performing a one way hash function on the keys associated with the affected intermediate nodes.

30. (Original) The method as recited in Claim 24, further comprising addressing the plurality of group controllers using a single virtual address.

31. (Previously Presented) A computer system that can manage addition and deletion of network nodes from and to a secure multicast or broadcast group of network nodes in a communications network without a single point of failure, wherein each of the network nodes is associated with one of a plurality of group controllers, wherein each group controller of the plurality of group controllers is a replica of a particular group controller, and wherein the network nodes and the plurality of group controllers are logically organized in a binary tree that represents the network nodes and the plurality of group controllers, in which leaf nodes of the binary tree represent network nodes that are joining or leaving the secure multicast or broadcast group, intermediate nodes represent other network nodes, and root nodes represent the plurality of group controllers, the computer system comprising:

a load balancer coupled to the plurality of group controllers for interfacing inbound service requests to a selected group controller of the plurality of group controllers;

a bus coupled to the load balancer for transferring data;

one or more processors coupled to the bus for selectively generating a group session key under control of program instructions;

a memory coupled to the one or more processors via the bus;

one or more sequences of program instructions stored in the memory which, when executed by the one or more processors cause the one or more processors to perform the steps of:

joining a first group controller to the plurality of group controllers in a local network;

establishing a secure communication channel between the first group controller and a second group controller of the plurality of group controllers using a key exchange protocol;

receiving a request to add or delete a network node of the secure multicast or broadcast group from the load balancer that is coupled to the plurality of group controllers;

creating and storing a new group session key for each network node represented in each branch of the binary tree that is affected by adding or deleting the network node from the secure multicast or broadcast group;

distributing the group session key from a third group controller of the plurality of group controllers to the network nodes.

32. (Previously Presented) A computer system as recited in Claim 31, wherein distributing a group session key further comprises:
receiving a token value at the third group controller to designate the third group controller as having permission to selectively generate the group session key and to generate node keys associated with the intermediate nodes and the leaf nodes; and
creating and storing the group session key only when the third group controller has the token value.

33. (Previously Presented) A computer system as recited in Claim 31, wherein distributing a group session key further comprises:
determining whether the secure multicast or broadcast group has a network node that is leaving the secure multicast or broadcast group;
determining which of the intermediate nodes are affected by the leaving node;
updating keys associated with the affected intermediate nodes;
generating a new group session key; and
sending the new group session key to the leaf nodes.

34. (Previously Presented) A computer system as recited in Claim 33, wherein updating keys comprises:
generating a new key of a parent node of the leaving node; and
encrypting the new key of the parent node with a key of a network node adjacent to the parent node.

35. (Previously Presented) A computer system as recited in Claim 31, wherein distributing a group session key further comprises:
receiving a request message from one of the network nodes to join the secure multicast or broadcast group;
determining which of the intermediate nodes are affected by the joining node;
updating keys associated with the affected intermediate nodes;
generating a new group session key and a private key of the joining node; and
sending a message comprising the new group session key, the private key, and the updated keys of affected intermediate nodes to the joining node.

36. (Previously Presented) A computer system as recited in Claim 35, wherein updating keys comprises performing a one way hash function on the keys associated with the affected intermediate nodes.

37. (Previously Presented) A computer system as recited in Claim 31, wherein receiving a request comprises receiving the request at a load balancer having a single virtual address that represents the plurality of group controllers.

38. (Previously Presented) A computer system as recited in Claim 37, further comprising one or more sequences of program instructions stored in the memory which, when executed by the one or more processors cause the one or more processors to perform the step of load balancing network traffic that is directed from a plurality of the network nodes to the plurality of group controllers.

39. (Previously Presented) A computer system as recited in Claim 31, wherein establishing a secure communication channel comprises exchanging a public key of the first group controller with all other group controllers in the plurality of group controllers based upon optimized broadcast Diffie-Hellman protocol.

40. (Previously Presented) A computer system as recited in Claim 35, wherein establishing a secure communication channel comprises:
receiving a public key value that is broadcast by the joining node;
sending a collective public key value from the network nodes to the joining node;
computing a shared secret key; and
creating and storing a group shared secret key by exchanging private key values.

41. (Previously Presented) An apparatus for managing addition and deletion of network nodes from and to a secure multicast or broadcast group of network nodes in a communications network without a single point of failure, wherein each of the network nodes is associated with one of a plurality of group controllers, wherein each group controller of the plurality of group controllers is a replica of a particular group controller, and wherein the network nodes and the plurality of group controllers are logically organized in a binary tree that represents the network nodes and the plurality of group controllers, in which leaf nodes of the binary tree represent network nodes that are joining or leaving the secure multicast or broadcast group, intermediate nodes represent other network nodes, and root nodes represent the plurality of group controllers, the apparatus comprising:

means for joining a first group controller to the plurality of group controllers in a local network;

means for establishing a secure communication channel between the first group controller and a second group controller of the plurality of group controllers using a key exchange protocol;

means for receiving a request to add or delete a network node of the secure multicast or broadcast group from a load balancer that is coupled to the plurality of group controllers;

means for creating and storing a new group session key for each network node represented in each branch of the binary tree that is affected by adding or deleting the network node from the secure multicast or broadcast group; and

means for distributing a group session key from a third group controller of the plurality of group controllers to the network nodes.

42. (Previously Presented) An apparatus as recited in Claim 41, wherein the means for distributing a group session key further comprises:

means for receiving a token value at the third group controller to designate the third group controller as having permission to selectively generate the group session key and to generate node keys associated with the intermediate nodes and the leaf nodes; and

means for creating and storing the group session key only when the third group controller has the token value.

43. (Previously Presented) An apparatus as recited in Claim 41, wherein the means for distributing a group session key further comprises:
means for determining whether the secure multicast or broadcast group has a network node that is leaving the secure multicast or broadcast group;
means for determining which of the intermediate nodes are affected by the leaving node;
means for updating keys associated with the affected intermediate nodes;
means for generating a new group session key; and
means for sending the new group session key to the leaf nodes.

44. (Previously Presented) An apparatus as recited in Claim 43, wherein the means for updating keys comprises:
means for generating a new key of a parent node of the leaving node; and
means for encrypting the new key of the parent node with a key of a network node adjacent to the parent node.

45. (Previously Presented) An apparatus as recited in Claim 41, wherein the means for distributing a group session key further comprises:
means for receiving a request message from one of the network nodes to join the secure multicast or broadcast group;
means for determining which of the intermediate nodes are affected by the joining node;
means for updating keys associated with the affected intermediate nodes;
means for generating a new group session key and a private key of the joining node; and
means for sending a message comprising the new group session key, the private key, and the updated keys of affected intermediate nodes to the joining node.

46. (Previously Presented) An apparatus as recited in Claim 45, wherein the means for updating keys comprises means for performing a one way hash function on the keys associated with the affected intermediate nodes.

47. (Previously Presented) An apparatus as recited in Claim 41, wherein the means for receiving a request comprises means for receiving the request at a load balancer having a single virtual address that represents the plurality of group controllers.

48. (Previously Presented) An apparatus as recited in Claim 47, further comprising means for load balancing network traffic that is directed from a plurality of the network nodes to the plurality of group controllers.

49. (Previously Presented) An apparatus as recited in Claim 41, wherein the means for establishing a secure communication channel comprises means for exchanging a public key of the first group controller with all other group controllers in the plurality of group controllers based upon optimized broadcast Diffie-Hellman protocol.

50. (Previously Presented) An apparatus as recited in Claim 45, wherein the means for establishing a secure communication channel comprises:
means for receiving a public key value that is broadcast by the joining node;
means for sending a collective public key value from the network nodes to the joining node;
means for computing a shared secret key; and
means for creating and storing a group shared secret key by exchanging private key values.

51. (Previously Presented) A computer-readable medium comprising one or more sequences of instructions for managing addition and deletion of network nodes from and to a secure multicast or broadcast group of network nodes in a communications network, wherein each of the network nodes is associated with a first group controller comprising information that is replicated in a plurality of group controllers, and wherein the network nodes and the plurality of group controllers are logically organized in a binary tree that represents the network nodes and the plurality of group controllers, in which leaf nodes of the binary tree represent network nodes that are joining or leaving the secure multicast or broadcast group, intermediate nodes represent other network nodes, and root nodes represent the plurality of group controllers, and which instructions, when executed by one or more processors, cause the processors to carry out the steps of:

joining the first group controller in a local network in which the plurality of group controllers are coupled;

establishing a secure channel between the first group controller and the plurality of group controllers through secure key exchange;

receiving a request to add or delete a network node from a load balancer that controls distribution of requests to the plurality of group controllers;

generating a new group session key for each network node represented in each branch of the binary tree that is affected by adding or deleting the network node from the secure multicast or broadcast group; and

distributing the group session key from the first group controller to the other group controllers of the plurality of group controllers over the secure channel.

52. (Previously Presented) A computer-readable medium as recited in Claim 51, further comprising instructions to carry out the step of generating the group session key only when the first group controller is designated as a master group controller that is authorized to join network nodes and generate group session keys.

53. (Previously Presented) A computer-readable medium as recited in Claim 52, further comprising instructions for carrying out the step of successively designating different group controllers of the plurality of group controllers as the master group controller in real time.

54. (Previously Presented) A computer-readable medium comprising one or more sequences of instructions for creating a secure multicast or broadcast group, and which instructions, when executed by one or more processors, cause the processors to carry out the steps of:

establishing a secure communication channel among a plurality of group controllers via a public key exchange protocol;

load balancing traffic emanating from a plurality of network nodes to the plurality of group controllers; and

distributing a group session key by one of the group controllers based upon a logical arrangement of the network nodes in a binary tree structure, the binary tree structure having a root node, intermediate nodes, and leaf nodes, wherein the plurality of network nodes correspond to leaf nodes of the binary tree structure and the plurality of group controllers correspond to the root node.

55. (Previously Presented) The computer-readable medium as recited in Claim 54, wherein the step of distributing further comprises:

circulating a token among the plurality of group controllers to designate the one group controller as having permission to selectively generate the group session key and keys associated with the intermediate nodes and the leaf nodes; and

selectively generating the group session key based upon the circulating step.

56. (Previously Presented) The computer-readable medium as recited in Claim 54, wherein the step of distributing further comprises:

detecting whether the secure multicast or broadcast group has a network node that is leaving the secure multicast or broadcast group;

determining which of the intermediate nodes are affected in response to the detecting step;

updating keys associated with the affected intermediate nodes;

generating a new group session key; and

sending the new group session key to the leaf nodes.

57. (Previously Presented) The computer-readable medium as recited in Claim 56, wherein the step of updating comprises:

generating a new key of a parent node of the leaving node; and

encrypting the new key of the parent node with a key of a network node adjacent to the parent node.

58. (Previously Presented) The computer-readable medium as recited in Claim 54, wherein the step of distributing further comprises:

receiving a request message from one of the plurality of network nodes to join the secure multicast or broadcast group;

determining which of the intermediate nodes are affected in response to the receiving step;

updating keys associated with the affected intermediate nodes;

generating a new group session key and a private key of the joining node; and

sending a message comprising the new group session key, the private key, and the updated keys of affected intermediate nodes to the joining node.

59. (Previously Presented) The computer-readable medium as recited in Claim 58, wherein the step of updating comprises performing a one way hash function on the keys associated with the affected intermediate nodes.

60. (Previously Presented) The computer-readable medium as recited in Claim 54, further comprising instructions for carrying out the step of addressing the plurality of group controllers using a single virtual address.

61. (Previously Presented) A computer system that can manage addition and deletion of network nodes from and to a secure multicast or broadcast group of network nodes in a communications network, wherein each of the network nodes is associated with a first group controller comprising information that is replicated in a plurality of group controllers, and wherein the network nodes and the plurality of group controllers are logically organized in a binary tree that represents the network nodes and the plurality of group controllers, in which leaf nodes of the binary tree represent network nodes that are joining or leaving the secure multicast or broadcast group, intermediate nodes represent other network nodes, and root nodes represent the plurality of group controllers, the computer system comprising:

a load balancer coupled to the plurality of group controllers for interfacing inbound service requests to a selected group controller of the plurality of group controllers;

a bus coupled to the load balancer for transferring data;

one or more processors coupled to the bus for selectively generating a group session key under control of program instructions;

a memory coupled to the one or more processors via the bus;

one or more sequences of program instructions stored in the memory which, when executed by the one or more processors cause the one or more processors to perform the steps of:

joining the first group controller in a local network in which the plurality of group controllers are coupled;

establishing a secure channel between the first group controller and the plurality of group controllers through secure key exchange;

receiving a request to add or delete a network node from the load balancer that controls distribution of requests to the plurality of group controllers;

generating a new group session key for each network node represented in each branch of the binary tree that is affected by adding or deleting the network node from the secure multicast or broadcast group; and

distributing the group session key from the first group controller to the other group controllers of the plurality of group controllers over the secure channel.

62. (Previously Presented) A computer system as recited in Claim 61, further comprising instructions to perform the step of generating the group session key only when the first group controller is designated as a master group controller that is authorized to join network nodes and generate group session keys.

63. (Previously Presented) A computer system as recited in Claim 62, further comprising instructions to perform the step of successively designating different group controllers of the plurality of group controllers as the master group controller in real time.

64. (Previously Presented) A computer system that can create a secure multicast or broadcast group, the computer system comprising:

a load balancer coupled to the plurality of group controllers for interfacing inbound service requests to a selected group controller of the plurality of group controllers;

a bus coupled to the load balancer for transferring data;

one or more processors coupled to the bus for selectively generating a group session key under control of program instructions;

a memory coupled to the one or more processors via the bus;

one or more sequences of program instructions stored in the memory which, when executed by the one or more processors cause the one or more processors to perform the steps of:

establishing a secure communication channel among a plurality of group controllers via a public key exchange protocol;

load balancing traffic emanating from a plurality of network nodes to the plurality of group controllers; and

distributing a group session key by one of the group controllers based upon a logical arrangement of the network nodes in a binary tree structure, the binary tree structure having a root node, intermediate nodes, and leaf nodes, wherein the plurality of network nodes correspond to leaf nodes of the binary tree structure and the plurality of group controllers correspond to the root node.

65. (Previously Presented) The computer system as recited in Claim 64, wherein the step of distributing further comprises:
circulating a token among the plurality of group controllers to designate the one group controller as having permission to selectively generate the group session key and keys associated with the intermediate nodes and the leaf nodes; and
selectively generating the group session key based upon the circulating step.

66. (Previously Presented) The computer system as recited in Claim 64, wherein the step of distributing further comprises:
detecting whether the secure multicast or broadcast group has a network node that is leaving the secure multicast or broadcast group;
determining which of the intermediate nodes are affected in response to the detecting step;
updating keys associated with the affected intermediate nodes;
generating a new group session key; and
sending the new group session key to the leaf nodes.

67. (Previously Presented) The computer system as recited in Claim 66, wherein the step of updating comprises:
generating a new key of a parent node of the leaving node; and
encrypting the new key of the parent node with a key of a network node adjacent to the parent node.

68. (Previously Presented) The computer system as recited in Claim 64, wherein the step of distributing further comprises:
receiving a request message from one of the plurality of network nodes to join the secure multicast or broadcast group;
determining which of the intermediate nodes are affected in response to the receiving step;
updating keys associated with the affected intermediate nodes;
generating a new group session key and a private key of the joining node; and
sending a message comprising the new group session key, the private key, and the updated keys of affected intermediate nodes to the joining node.

69. (Previously Presented) The computer system as recited in Claim 68, wherein the step of updating comprises performing a one way hash function on the keys associated with the affected intermediate nodes.

70. (Previously Presented) The computer system as recited in Claim 64, further comprising instructions to perform the step of addressing the plurality of group controllers using a single virtual address.

71. (Previously Presented) An apparatus for managing addition and deletion of network nodes from and to a secure multicast or broadcast group of network nodes in a communications network, wherein each of the network nodes is associated with a first group controller comprising information that is replicated in a plurality of group controllers, and wherein the network nodes and the plurality of group controllers are logically organized in a binary tree that represents the network nodes and the plurality of group controllers, in which leaf nodes of the binary tree represent network nodes that are joining or leaving the secure multicast or broadcast group, intermediate nodes represent other network nodes, and root nodes represent the plurality of group controllers, the apparatus comprising:

means for joining the first group controller in a local network in which the plurality of group controllers are coupled;

means for establishing a secure channel between the first group controller and the plurality of group controllers through secure key exchange;

means for receiving a request to add or delete a network node from a load balancer that controls distribution of requests to the plurality of group controllers;

means for generating a new group session key for each network node represented in each branch of the binary tree that is affected by adding or deleting the network node from the secure multicast or broadcast group; and

means for distributing the group session key from the first group controller to the other group controllers of the plurality of group controllers over the secure channel.

72. (Previously Presented) An apparatus as recited in Claim 71, further comprising means for generating the group session key only when the first group controller is designated as a master group controller that is authorized to join network nodes and generate group session keys.

73. (Previously Presented) An apparatus as recited in Claim 72, further comprising means for successively designating different group controllers of the plurality of group controllers as the master group controller in real time.

74. (Previously Presented) An apparatus for creating a secure multicast or broadcast group, the apparatus comprising:

means for establishing a secure communication channel among a plurality of group controllers via a public key exchange protocol;

means for load balancing traffic emanating from a plurality of network nodes to the plurality of group controllers; and

means for distributing a group session key by one of the group controllers based upon a logical arrangement of the network nodes in a binary tree structure, the binary tree structure having a root node, intermediate nodes, and leaf nodes, wherein the plurality of network nodes correspond to leaf nodes of the binary tree structure and the plurality of group controllers correspond to the root node.

75. (Previously Presented) The apparatus as recited in Claim 74, wherein the means for distributing further comprises:

means for circulating a token among the plurality of group controllers to designate the one group controller as having permission to selectively generate the group session key and keys associated with the intermediate nodes and the leaf nodes; and

means for selectively generating the group session key based upon the circulating step.

76. (Previously Presented) The apparatus as recited in Claim 74, wherein the means for distributing further comprises:

means for detecting whether the secure multicast or broadcast group has a network node that is leaving the secure multicast or broadcast group;

means for determining which of the intermediate nodes are affected in response to the detecting means;

means for updating keys associated with the affected intermediate nodes;

means for generating a new group session key; and

means for sending the new group session key to the leaf nodes.

77. (Previously Presented) The apparatus as recited in Claim 76, wherein the means for updating comprises:

means for generating a new key of a parent node of the leaving node; and

means for encrypting the new key of the parent node with a key of a network node adjacent to the parent node.

78. (Previously Presented) The apparatus as recited in Claim 74, wherein the means for distributing further comprises:

means for receiving a request message from one of the plurality of network nodes to join the secure multicast or broadcast group;

means for determining which of the intermediate nodes are affected in response to the receiving means;

means for updating keys associated with the affected intermediate nodes;

means for generating a new group session key and a private key of the joining node; and

means for sending a message comprising the new group session key, the private key, and the updated keys of affected intermediate nodes to the joining node.

79. (Previously Presented) The apparatus as recited in Claim 78, wherein the means for updating comprises means for performing a one way hash function on the keys associated with the affected intermediate nodes.

80. (Previously Presented) The apparatus as recited in Claim 74, further comprising means for addressing the plurality of group controllers using a single virtual address.